

OpenStack Private Cloud Design and Cost Model for Real Estate Data Management

Reynaldo^{1)*}, Alfa Ryano Yohannis²⁾

^{1,2)}Department of Information Technology, Pradita University, Indonesia
Scientia Business Park, Jalan Gading Serpong Boulevard, Tangerang 15810, Indonesia

¹⁾reynaldo@student.pradita.ac.id, ²⁾alfa.ryano@pradita.ac.id

Submitted : Sep 5, 2026 | **Accepted** : Oct 5, 2026 | **Published** : Oct 5, 2026

Abstract: Large real estate enterprises hold dense concentrations of regulated data while running enterprise applications at steady, predictable load, a combination poorly matched to public cloud pricing and to cross-border processing under Indonesia's Personal Data Protection Law. This study aims to propose an on-premises private cloud design for such an organisation and to model its cost position with all assumptions explicitly stated. The method comprises four parts: a real estate workload profile that maps application classes to criticality labels; a four-layer reference architecture built on OpenStack with a Ceph storage backend and an orchestration layer adapted from the Private Cloud Bespoke Orchestrator framework, in which a single data-criticality label propagates through storage placement, network isolation, snapshot cadence, and recovery ordering; an analytical estimate and a stochastic simulation of disaster recovery duration under three load conditions; and a parametric three-year total cost of ownership model that includes transition costs, evaluated through three scenarios and a Monte Carlo analysis. In the reference scenario, the private cloud becomes cost-favourable once annual public cloud expenditure exceeds approximately IDR 393 million, with a 90 per cent interval of IDR 272 to 600 million. Post-migration staffing and node acquisition cost are the dominant model assumptions. In simulation, recovering 1.0 terabyte of critical data meets the thirty-minute target in 63 per cent of runs at moderate load. The design has not been deployed or audited, so simulated recovery times rest on assumed parameters and the compliance posture is unverified.

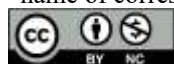
Keywords: break-even analysis; Ceph; disaster recovery; OpenStack; private cloud; total cost of ownership

INTRODUCTION

Real estate enterprises hold an unusually dense concentration of sensitive records. A single large developer maintains client identity documents, transaction histories, financing arrangements, title deeds, and lease agreements, most of which now exist only in digital form. A systematic review of digital transformation in the sector synthesises thirty-six studies and identifies infrastructure cost and cybersecurity exposure as the two most frequently reported barriers to adoption (Al-haimi et al., 2025). For an organisation that already operates a data centre, the practical question is not whether to adopt cloud service models but which deployment model reconciles cost, control, and regulatory obligation.

Two properties of the enterprise workload bear on that choice. The first is load stability: customer relationship management, enterprise resource planning, and document management systems run continuously at predictable utilisation, so an organisation purchasing elastic capacity pays a permanent premium for flexibility it never exercises. The second is jurisdiction. Law Number 27 of 2022 on Personal Data Protection imposes obligations on lawful processing, security, and cross-border transfer (Republic of Indonesia, 2022). Its two-year transition concluded in October 2024, and the compliance burden falls most heavily on organisations without a dedicated data protection function (Syailendra &

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Lie, 2024). Retaining processing under direct organisational control can therefore ease the demonstration of compliance, although compliance itself must be established by assessment and cannot be inferred from the deployment model.

Private cloud infrastructure provides on-demand provisioning, virtual machine lifecycle management, and storage scalability within the organisation's own facility, matching the model provisioned for exclusive use by a single organisation (Mell & Grance, 2011). OpenStack is the dominant open-source platform for such deployments, with viability established through fault-resistance studies (Heuchert et al., 2025) and multinode evaluation (Ciptaningtyas et al., 2023).

Three gaps motivate this work. First, published OpenStack implementations concentrate on educational and research settings (Zhao et al., 2024; Ciptaningtyas et al., 2023) or generic small-scale infrastructure as a service (Heuchert et al., 2025), whereas the real estate domain is addressed at the level of platform and market analysis (Tagliaro et al., 2025; Harth et al., 2025). Second, orchestration and disaster recovery are studied separately: Park et al. (2025) automate bespoke virtual machine environments without a recovery objective regime, while recovery studies establish metrics without an orchestration layer (Suyatno et al., 2025; Nawrocki & Smendowski, 2025). Third, published cost comparisons report single savings percentages without exposing their assumptions or the transition costs preceding any saving.

This study addresses these gaps through design and analysis rather than production deployment, so its architecture is a proposal and not a validated system. Its objectives are to characterise the workload of a real estate enterprise and derive a criticality mapping from it; to specify an architecture in which regulatory data classification is expressed as infrastructure configuration, with a reference hardware configuration, a deployment procedure, and an analytical estimate and stochastic simulation of recovery duration; and to construct a parametric cost model whose every input is named and whose scope covers transition costs. This paper presents no measured provisioning latency or recovery time and no compliance assessment, and identifies the empirical work required to close that gap.

LITERATURE REVIEW

Liu et al. (2025) formulate private cloud deployment model selection as a cost optimisation problem and find that the allocation of ownership and maintenance responsibility is the dominant lever on total infrastructure cost. This result motivates the cost decomposition adopted here. Buyya et al. (2024) set out directions for integrated management of data centre resources and workloads, arguing that energy and sustainability considerations must be treated jointly with resource allocation rather than as an afterthought.

On platform viability, Heuchert et al. (2025) drive a five-node OpenStack deployment to failure and show that a multinode configuration sustains resiliency and elasticity within defined boundaries. Ciptaningtyas et al. (2023) deploy OpenStack through Kolla-Ansible at an Indonesian university and report that per-instance processor throughput declines by 1.41 per cent and memory throughput by 4.46 per cent as concurrent instances rise, while creation latency is largely insensitive. Zhao et al. (2024) pair OpenStack with Ceph and report unified management at reduced cost. These establish platform viability at this scale, but none evaluates recovery objectives together with cost of ownership.

Park et al. (2025) introduce the Private Cloud Bespoke Orchestrator, which automates construction, operation, and recovery of bespoke virtual machine environments on OpenStack and reduces manual administrative effort. The framework is the methodological basis of the architecture proposed here. In its published form it applies a uniform policy across virtual machine classes; the adaptation described in the following section instead derives policy from application data criticality.

Work on resource management supplies the operational context for consolidation: Zou et al. (2024) observe that migration trades energy saving against service level violation, Goyal and Awasthi (2025) that excessive utilisation degrades performance, Banerjee et al. (2024) and Yao et al. (2023) treat multi-resource and load-balancing consolidation, Lambert et al. (2026) report persistent underutilisation from overprovisioning, and Hidayat et al. (2025) apply reinforcement learning to hybrid migration. Khoda Parast et al. (2025) quantify encryption overhead at the Ceph layer.

On recovery, recovery time objective is the maximum period a resource may remain unavailable before impact becomes unacceptable, and recovery point objective is the point to which data can be

recovered from the most recent backup (Swanson et al., 2010). A recovery point objective as configured is a design parameter, whereas data actually lost is a measurement. Suyatno et al. (2025) observe that recovery duration is dominated by transfer volume rather than orchestration logic, and Nawrocki and Smendowski (2025) find that automated orchestration reduces it principally by removing human latency from the failure response path.

On the application domain, Starr et al. (2021) frame the convergence of industrial technologies on commercial real estate, Tagliaro et al. (2025) examine what the sector requires from technology, and Harth et al. (2025) cluster emerging technologies into a framework guiding resource allocation. None addresses the infrastructure substrate on which real estate applications execute.

Table 1 positions the present study against this body of work along five dimensions. No prior study combines orchestration in the manner of the Private Cloud Bespoke Orchestrator, a specified recovery regime, and a parametric cost model within the real estate domain.

Table 1. Positioning of the present study relative to prior work

Study	Platform basis	VM automation	Recovery	Cost	Domain
Liu et al. (2025)	Analytical	No	No	Model	Generic
Park et al. (2025)	OpenStack	Yes	Partial	No	Generic
Zhao et al. (2024)	OpenStack, Ceph	No	No	Partial	Education
Heuchert et al. (2025)	OpenStack	No	Empirical	No	Small IaaS
Ciptaningtyas et al. (2023)	OpenStack, Kolla	No	No	No	University
Suyatno et al. (2025)	Hybrid	No	Empirical	No	Generic
Nawrocki & Smendowski (2025)	Survey	Yes	Partial	Qualitative	Generic
Al-haimi et al. (2025)	None, review	No	No	No	Real estate
This study	OpenStack, Ceph, PCBO	Yes	Design, simulation	Parametric	Real estate

METHOD

The study proceeds in four stages: workload characterisation, architectural specification, deployment specification, and construction of a parametric cost model. The orchestration approach follows Park et al. (2025), the only published framework that treats construction, operation, and recovery of OpenStack virtual machine environments as a single automated concern; recovery objectives follow Swanson et al. (2010) and cost decomposition follows Liu et al. (2025).

Data used in the cost model are of two kinds. Externally sourced values are the electricity tariff for class B-3 medium-voltage business customers above 200 kVA (Ministry of Energy and Mineral Resources of the Republic of Indonesia, 2024) and the three-year amortisation horizon following Liu et al. (2025). Node power draw is an engineering estimate from component nameplate envelopes. All other values, including every transition cost item, are indicative, stated so the worked example is reproducible and varied in the scenario and Monte Carlo analyses; they are not the prices faced by any particular organisation.

Real Estate Workload Profile and Criticality Mapping

Real estate data is document-centric and long-lived: each transaction produces identity, financing, and legal documents that must stay retrievable for the life of the contract, so archives dominate volume but change rarely. Load peaks are calendar-driven (project launches, instalment billing, month-end closing) and plannable. Financial records must be kept ten years from the end of the financial year, while other company documents follow their usefulness (Republic of Indonesia, 1997), and buyer and tenant records are personal data under Law Number 27 of 2022 (Republic of Indonesia, 2022). Sales, booking, and billing tolerate little downtime, whereas archives need integrity more than minute-level availability.

Table 2 maps five application classes to three criticality labels. The volumes and targets are design assumptions obtained by decomposing the 8 TB working set, not company measurements, and should be replaced by an organisation's own inventory and business impact analysis.

Table 2. Real estate workload classes, criticality labels, and design targets

Application class	Representative data	Label	Assumed volume	Availability, RPO, RTO targets	Retention and regulatory basis
Client and transaction (CRM, booking, payment)	Identity, bookings, instalments	Critical	0.4 TB	99.9 per cent, 60 min, 30 min	Personal data, Law 27/2022

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Application class	Representative data	Label	Assumed volume	Availability, RPO, RTO targets	Retention and regulatory basis
Financial ERP	Ledgers, receivables, project costing	Critical	0.6 TB	99.9 per cent, 60 min, 30 min	10 years, Law 8/1997
Legal and title documents (document management)	Sale agreements, title deed scans, leases	Important	3.6 TB	99.5 per cent, 4 h, 4 h	Company schedule, long term
Project and construction documents	Drawings, reports, permits	Important	1.4 TB	99.5 per cent, 4 h, 4 h	Company schedule
Derived and reconstructible data	Analytics extracts, marketing assets	Standard	2.0 TB	99.0 per cent, 24 h, 24 h	Regenerated from sources

The critical class totals 1.0 TB, the reference volume of the recovery estimate below. Replication factor three with daily snapshot verification applies to every label, because archived documents change rarely and their loss is irreversible.

Reference Architecture

The architecture comprises four layers: physical infrastructure, OpenStack platform, orchestration, and real estate application services. Fig. 1 shows the layering and the principal interfaces between layers.

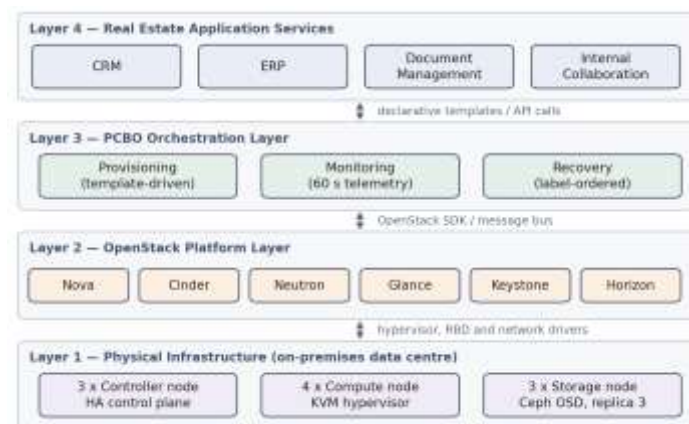


Fig. 1 Four-layer reference architecture for real estate private cloud.

Nodes take three roles. Three controller nodes host the control plane, so quorum-based services tolerate the loss of one. Four compute nodes host tenant virtual machines under the KVM hypervisor. Three storage nodes run Ceph object storage daemons, three being the minimum permitting replication factor three with a host-level failure domain. A 25 Gigabit Ethernet fabric carries management, tenant, and storage replication traffic on separate virtual local area networks with redundant switching; replication is isolated because recovery and tenant traffic contend directly, and the transfer-bound character of recovery (Suyatno et al., 2025) means contention extends recovery duration.

Table 3 states the reference configuration. Registered error-correcting memory is specified throughout because the workload persists client identity and transaction records, and silent memory corruption in a storage or hypervisor node is not detectable at the application layer.

Table 3. Reference hardware configuration

Role	Quantity	Processor	Memory	Storage	Network
Controller	3	2 x 12-core server class, 2.0 GHz base or higher	128 GB DDR5 ECC RDIMM	2 x 480 GB SATA SSD, RAID 1, operating system	2 x 25 GbE, BMC
Compute	4	2 x 16-core server class, 2.5 GHz base or higher	512 GB DDR5 ECC RDIMM	2 x 480 GB SATA SSD, RAID 1, operating system	2 x 25 GbE, BMC
Storage	3	1 x 12-core server class	128 GB DDR5 ECC RDIMM	2 x 480 GB SSD, operating system; 8 x 3.84 TB SSD, OSD	2 x 25 GbE, BMC
Switch	2	Not applicable	Not applicable	Not applicable	48 x 25 GbE, 4 x 100 GbE uplink

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

The configuration provides 128 physical compute cores and two terabytes of compute memory. Ceph raw capacity is 92.2 terabytes across twenty-four devices; at replication factor three with twenty per cent rebalancing headroom, usable capacity is approximately 24.6 terabytes. This is sized against a stated design assumption rather than a measured inventory: an initial working set of eight terabytes (Table 2), twenty-five per cent annual growth giving 15.6 terabytes over the horizon, and a thirty per cent snapshot allowance, yielding 20.3 terabytes and a margin near eighteen per cent.

Six OpenStack services are specified. Nova schedules instances and exposes the evacuation interface; Cinder supplies Ceph-backed block volumes with snapshots; Neutron manages tenant networks and security groups over Open vSwitch; Glance stores templates, Keystone enforces role-based access control, and Horizon provides administration.

Orchestration and Policy Derivation

The orchestration layer implements automated environment construction, continuous monitoring, and recovery on failure detection, shown in Fig. 2. Construction reads a declarative template and concludes with a post-boot health probe before the instance is reported ready. Monitoring polls instance state and telemetry against per-template thresholds. Recovery handles failure at instance and node level: a failed instance is rebuilt from its most recent verified snapshot, while a failed compute node triggers evacuation of resident instances to surviving nodes.

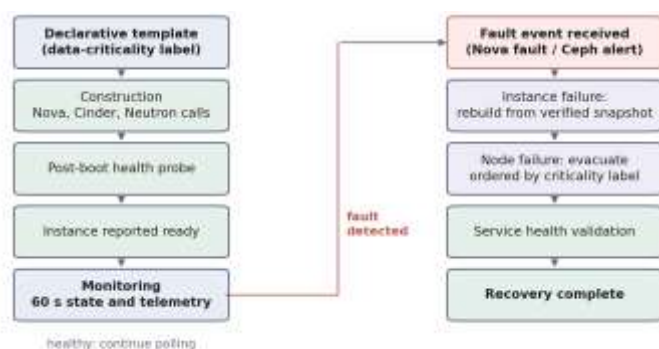


Fig. 2 Orchestration and recovery control flow.

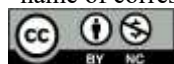
The adaptation proposed here concerns policy derivation. In the framework as published, snapshot cadence and recovery ordering are properties of the virtual machine class. In this design, they are derived from a data-criticality label attached to the application the virtual machine serves. Workloads labelled critical, which hold client identity, transaction, and financial records, receive the shortest snapshot interval and the highest evacuation priority; workloads labelled important follow, and workloads labelled standard, which hold derived or reconstructible data, receive the longest intervals and are evacuated last. The label is declared once in the template and is the same label that determines Ceph pool placement and Neutron network assignment, so a single classification decision propagates consistently through storage, network, and recovery policy.

The intended consequence is auditability. Under a uniform policy, an auditor asking which systems hold regulated data must reconstruct the answer from operational practice. Under label-derived policy, the answer is a property of declared configuration, and divergence between classification and treatment becomes a detectable configuration error rather than an undocumented convention. This supports, but does not prove, compliance with Law Number 27 of 2022.

Disaster Recovery Design and Analytical Bound

Design targets for workloads labelled critical are a recovery time objective of thirty minutes and a recovery point objective of sixty minutes; the lower labels follow Table 2. These are targets, not demonstrated capabilities. The design operates in three tiers, shown in Fig. 3: continuous replication addresses component loss but not logical corruption; scheduled snapshotting runs at the interval set by the label, retaining changed blocks, with a daily job that mounts each snapshot on a scratch instance, because an unverified snapshot is not a recovery point; and automated failover rebuilds affected instances from the latest verified snapshot on an alternate node in label order, validating service health before recovery is marked complete.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

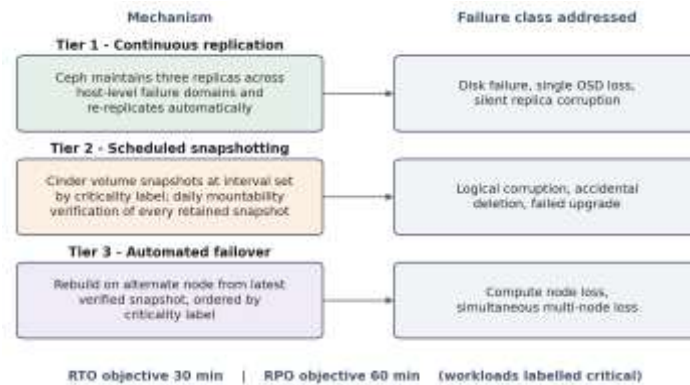


Fig. 3 Three-tier disaster recovery design and associated failure classes.

Because recovery duration is dominated by transfer volume rather than orchestration logic (Suyatno et al., 2025), the thirty-minute target can first be estimated analytically. Let V denote the critical data volume to be rehydrated and R the effective reconstruction throughput of the storage network. Estimated recovery duration is given by (1), in which T_{detect} is bounded by the monitoring poll interval, T_{sched} covers scheduling, evacuation initiation, and snapshot access, T_{boot} covers instance boot and network reconfiguration, and T_{validate} is bounded by the health probe timeout; storage contention is represented by a reduced R . Queueing among concurrent failures and cache warm-up remain unmodelled.

$$T = T_{\text{detect}} + T_{\text{sched}} + T_{\text{boot}} + V/R + T_{\text{validate}} \quad (1)$$

A 25 Gigabit Ethernet network has a line rate of about 3,125 megabytes per second. Table 4 evaluates (1) for sustained reconstruction throughput at one half, one quarter, and one tenth of line rate, representing light, moderate, and heavy tenant load, with assumed delays of 60 seconds for T_{detect} , 120 for T_{sched} , 180 for T_{boot} , and 60 for T_{validate} . At moderate load, recovering the 1.0 terabyte critical class takes approximately 28 minutes, inside the target by less than two minutes, whereas heavy load gives about 60 minutes. Because (1) is a point estimate that sums every delay in sequence, a stochastic simulation tests its robustness.

The simulation is a time-stepped stochastic model of the procedure above. Detection takes between 0 and 60 seconds because the node fails at a random phase of the poll; scheduling and boot with network reconfiguration are log-normal with means of 120 and 180 seconds; the health probe takes 20 to 60 seconds; at most five instances are rebuilt concurrently, critical ones first; and background tenant traffic follows a mean-reverting random process whose mean utilisation of 20, 60, and 84 per cent reproduces the three throughput levels. Fixed delays reproduce (1) to within 0.03 minutes. Each cell of Table 4 uses 5,000 runs with a fixed seed. The distributions are assumptions, so the output describes the design under them and is not a benchmark of OpenStack or Ceph.

Simulated means are about one minute above the analytic estimate under light and moderate load and below it under heavy load, where capping background utilisation at 95 per cent raises mean throughput slightly; the 95th percentile adds three to ten minutes. At moderate load the thirty-minute target is met in 63 per cent of runs for 1.0 terabyte and in every run for 0.5 terabyte; with 95 per cent probability it holds up to about 1.8, 0.9, and 0.4 terabytes under light, moderate, and heavy load. Limiting concurrency to two rebuilds raises the moderate-load mean for 1.0 terabyte from 29.4 to 32.0 minutes, and a 300-second mean boot raises it to 32.8. Recovering all labels at moderate load completes the critical, important, and standard classes after about 29 minutes, 2.3 hours, and 3.0 hours, so the target is not reliably met for 1.0 terabyte at moderate load.

Table 4. Recovery duration of the critical class in minutes: analytic estimate / simulated mean / simulated 95th percentile, and simulated probability of meeting the thirty-minute target

Load condition	V = 0.5 TB	V = 1.0 TB	V = 2.0 TB	P(30 min), V = 0.5 / 1.0 / 2.0	Largest V, P at least 0.95
Light (R = 1/2 of line rate)	12.3 / 13.6 / 16.2	17.7 / 18.8 / 21.5	28.3 / 29.4 / 32.2	1.00/1.00/0.68	1.79 TB

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Load condition	V = 0.5 TB	V = 1.0 TB	V = 2.0 TB	P(30 min), V = 0.5 / 1.0 / 2.0	Largest V, P at least 0.95
Moderate (R = 1/4 of line rate)	17.7 / 18.8 / 22.0	28.3 / 29.4 / 33.0	49.7 / 50.8 / 55.2	1.00/0.63/0.00	0.86 TB
Heavy (R = 1/10 of line rate)	33.7 / 31.9 / 36.8	60.3 / 55.5 / 62.0	113.7 / 102.6 / 112.2	0.27/0.00/0.00	0.37 TB

Deployment Specification

The deployment procedure is specified in order. All nodes run Ubuntu Server 22.04 LTS with a minimal base selection. Management, tenant, and storage networks are configured across redundant top-of-rack switches, and time synchronisation is verified cluster-wide before service deployment, because clock skew produces failures that present as unrelated authorisation errors.

OpenStack is deployed using Kolla-Ansible, selected over a single-node installer because the target is multinode with a high-availability control plane and because comparable Indonesian deployments report it as workable at this scale (Ciptaningtyas et al., 2023). The sequence is: Ceph bootstrap with replication factor three and a host-level failure domain; control plane across the three controllers; compute services and the networking agent; and the Ceph block driver for the volume and image services. The target release is OpenStack 2025.1 Epoxy, a Skip Level Upgrade Release Process release whose tested runtime includes the host operating system above.

The orchestrator runs as a service on the control plane and interacts with the platform through its software development kit. Provisioning reads templates from a version-controlled repository; monitoring polls instance state and telemetry at sixty-second intervals; recovery subscribes to instance fault events and storage health alerts. Snapshot interval is set per criticality label, retention is bounded at forty-eight snapshots per critical volume, and snapshots failing daily verification are regenerated.

Migration is specified in two stages. Non-critical services move first to establish platform stability under real load, each converted from physical to virtual form and registered in the image service. Mission-critical workloads follow in a planned maintenance window, with integrity verified by database record counts and application-level tests, and name resolution is repointed only after both pass, so rollback needs no propagation delay. Conversion labour and parallel-operation duration enter the cost model as C_{mig} and τ .

Cost of Ownership Model

Two scenarios are compared over three years. Scenario A is the pre-migration arrangement, in which enterprise applications run on legacy physical servers together with public cloud subscriptions. Scenario B consolidates workloads onto the private cloud of Table 3. Costs are decomposed into capital expenditure, operational expenditure, and subscription or licensing, held mutually exclusive because mixing operational subscription fees with capital acquisition in one line item renders the scenarios incomparable. The model covers acquisition, operation, and the transition items of migration labour, training, facility upgrade, parallel operation, and hardware refresh, so it is an extended total cost of ownership model for this scope.

Capital expenditure for Scenario B is the sum of node acquisition and network equipment, given in (2), where N is the number of server nodes, excluding switches, C_{node} is the acquisition cost per node, and C_{net} is the cost of switching and cabling.

$$C_{capex} = N C_{node} + C_{net} \quad (2)$$

Annual energy cost follows from aggregate nameplate draw, facility power usage effectiveness, and tariff, as given in (3), where n_r and P_r are the count and per-node draw of role r , PUE is the power usage effectiveness, H is the number of hours in a year, and T_e is the tariff per kilowatt-hour.

$$E_{year} = (SUM n_r P_r) \times PUE \times H \times T_e \quad (3)$$

One-time transition cost is given in (4), where C_{mig} is migration labour, C_{train} is training, C_{fac} is the facility upgrade for power distribution and cooling, and ρ is the fraction of capital expenditure reserved for hardware refresh and spares within the horizon.

$$C_{one} = C_{mig} + C_{train} + C_{fac} + \rho C_{capex} \quad (4)$$

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Annual maintenance is taken as a fraction m of capital expenditure. Total three-year cost for Scenario B is then given in (5), where Y is the horizon in years, F_B is post-migration infrastructure staffing in full-time equivalents, W is the annual cost per full-time equivalent, and L_B is any optional support subscription, and τ is the duration in years of parallel operation, during which the legacy estate and the public cloud subscription are paid alongside the new platform.

$$TCO_B = C_{capex} + C_{one} + Y(E_{year} + m C_{capex} + F_B W + L_B) + \tau(S_a + M_{leg} + E_{leg}) \quad (5)$$

Scenario A carries no new capital expenditure and is given in (6), where S_a is the annual public cloud subscription, M_{leg} and E_{leg} are legacy server maintenance and energy, and F_A is pre-migration staffing.

$$TCO_A = Y(S_a + M_{leg} + E_{leg} + F_A W) \quad (6)$$

Setting the two equal yields the break-even subscription, the annual public cloud expenditure at which the scenarios cost the same, given in (7), in which K_B is the cost of Scenario B in (5) without its parallel-operation term.

$$S_a^* = (K_B - (Y - \tau)(M_{leg} + E_{leg}) - Y F_A W) / (Y - \tau) \quad (7)$$

Reporting a break-even threshold rather than a savings percentage, which is a single point conditioned on one assumed subscription level, lets any organisation compare its own invoice. Table 5 states the parameter values of three scenarios. The optimistic and pessimistic scenarios set every cost parameter at the end of its range that favours or disfavors the private cloud, so they bound the result rather than predict it. A Monte Carlo analysis draws every parameter independently from a triangular distribution over the same range (100,000 draws), varying W from IDR 150 to 220 million.

Table 5. Parameter values of the optimistic, reference, and pessimistic scenarios

Parameter	Description	Optimistic	Reference	Pessimistic	Basis
C_{node}, C_{net}	Cost per server node ($N = 10$, Table 3); switching and cabling, IDR million	60, 120	90, 150	120, 200	Indicative
$P_{ctrl}, P_{comp}, P_{stor}$	Per-node draw by role, W	350, 600, 450	350, 600, 450	350, 600, 450	Nameplate estimate
PUE	Facility power usage effectiveness	1.4	1.6	1.8	Typical enterprise room
T_e	Electricity tariff, class B-3, IDR per kWh	1,114.74	1,114.74	1,114.74	Ministry of Energy and Mineral Resources (2024)
m, ρ	Annual maintenance; refresh and spares reserve, per cent of capital	6, 5	8, 10	10, 15	Indicative
W	Loaded staff cost, IDR million per FTE-year	180	180	180	Indicative; varied in Monte Carlo
F_A, F_B	Staffing before and after migration, FTE	2.5, 1.0	2.5, 1.5	2.5, 2.5	Indicative
L_B	Optional support subscription, IDR million per year	0	50	100	Indicative
M_{leg}, E_{leg}	Legacy maintenance and energy, IDR million per year	140, 84	100, 60	70, 42	Indicative
Y	Amortisation horizon, years	3	3	3	Liu et al. (2025)
$C_{mig}, C_{train}, C_{fac}$	Migration labour, training, facility upgrade, IDR million	60, 30, 50	120, 60, 100	180, 90, 150	Indicative
τ	Parallel operation, months	2	3	6	Indicative
S_a^*	Break-even subscription, IDR million per year	-78	393	1,089	Equation (7)

RESULT

Under the reference scenario of Table 5, aggregate nameplate draw across the ten server nodes is 4.80 kilowatts, which at a power usage effectiveness of 1.6 gives 67,277 kilowatt-hours per year and an annual energy cost of IDR 75.0 million at the stated tariff. Capital expenditure is IDR 1,050 million, generating annual maintenance of IDR 84.0 million. The partial model sums to IDR 2,487 million over

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

three years. One-time transition costs add IDR 385 million (migration labour 120, training 60, facility upgrade 100, refresh reserve 105), giving IDR 2,872 million before the parallel-operation term. Staffing accounts for IDR 810 million, exceeding energy (IDR 225 million) and maintenance combined.

Substituting these values into (7) gives a break-even subscription of IDR 393 million per year, equivalent to approximately IDR 32.8 million per month. The partial model without the transition terms gives IDR 219 million, so omitting them understates the threshold by about 80 per cent. Fig. 4 shows the three-year cost of both scenarios as a function of the annual public cloud subscription, with the break-even points of the extended and partial models marked.

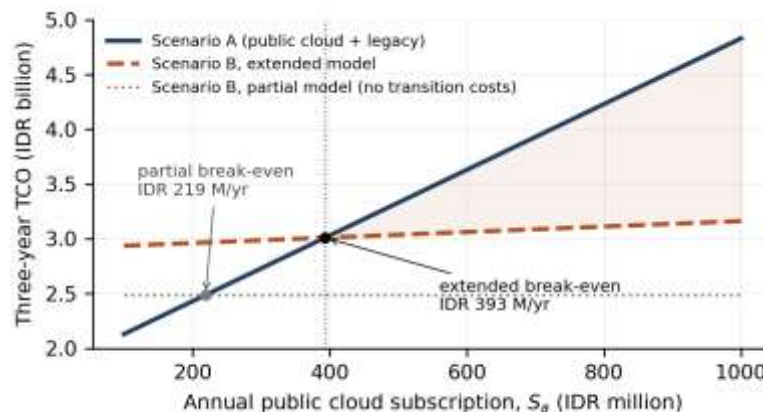


Fig. 4 Three-year cost of ownership for both scenarios against annual public cloud subscription, with break-even points of the partial and extended models.

Below the threshold, migration increases cost: the extended model gives a change of -21.9 per cent at a subscription of IDR 200 million and -9.4 per cent at IDR 300 million. Above it the reduction in three-year cost reaches 8.8 per cent at IDR 500 million and 34.5 per cent at IDR 1,000 million.

Holding staffing unchanged at 2.5 full-time equivalents moves the extended-model threshold from IDR 393 million to IDR 590 million per year, an increase of 50 per cent arising from a single assumption.

In Table 5 the threshold falls below zero in the optimistic scenario and rises to IDR 1,089 million per year in the pessimistic scenario. The Monte Carlo analysis gives a median of IDR 432 million per year and a 90 per cent interval of IDR 272 to 600 million; the probability of a threshold at or below the IDR 219 million of the partial model is 1.3 per cent. One at a time, post-migration staffing (1.0 to 2.5 full-time equivalents) and node cost (IDR 60 to 120 million) move the threshold by about IDR 295 million and IDR 292 million, whereas every other parameter moves it by less than IDR 115 million; their rank correlations with the threshold are 0.62 and 0.60, against at most 0.22 for any other parameter.

DISCUSSIONS

The results establish that cost advantage in this design is conditional rather than intrinsic. The negative reduction below the threshold, about 22 per cent at a subscription of IDR 200 million, is the most instructive result, because it identifies a region in which a migration justified on cost grounds would increase expenditure; a single positive savings percentage cannot express it. The scenario bounds, from below zero to IDR 1,089 million per year, show that the IDR 393 million threshold is a worked illustration that does not transfer to other organisations.

The sensitivity result cuts against the usual framing of orchestration benefit. The reduction from 2.5 to 1.5 full-time equivalents is a model assumption, not an observed outcome, and no evidence is offered that OpenStack removes one position. Staffing level and node acquisition cost are the two dominant sources of variation, so the result states what must hold for migration to pay, not that it will. Automation reduces cost only if it reduces staffing expenditure; where it redirects staff effort to other work, the benefit is released capacity that a cost of ownership calculation does not show.

Comparison with prior work supports the direction of the result while limiting its strength. The reductions at higher subscription levels are consistent with Liu et al. (2025), who find ownership and maintenance allocation dominant, although staffing as a model assumption cannot confirm that

mechanism. The capacity reasoning aligns with the consolidation literature (Goyal & Awasthi, 2025; Banerjee et al., 2024; Lambert et al., 2026), and the recovery results with the transfer-dominated behaviour reported by Suyatno et al. (2025) and the orchestration role described by Nawrocki and Smendowski (2025).

Heuchert et al. (2025) and Ciptaningtyas et al. (2023) show that OpenStack sustains this workload class and quantify contention under load, which the present study does not measure; neither addresses cost of ownership or recovery objectives, which it addresses by model and simulation.

Several limitations bound the results. This is a design and analytical exercise: no deployment is reported, and the simulated recovery times rest on assumed throughput and delay distributions. The contribution is therefore a proposed architecture and a model, not evidence that the architecture performs as designed; a minimal proof of concept measuring provisioning latency, node evacuation time, and Ceph reconstruction throughput under tenant load is the first step needed. Most cost parameters in Table 5 are indicative rather than sourced, and the scenario and Monte Carlo analyses expose their combined effect without removing the dependency. The model omits legacy decommissioning, security audit cost, downtime loss, and floor space charges, which favour Scenario B, and grants no residual hardware value, which favours Scenario A.

Further threats concern scope and unevaluated claims. The reference configuration assumes an existing data centre with spare power, cooling, and rack capacity. The workload profile of Table 2 rests on stated assumptions rather than a data inventory, so the link between the real estate domain and the design requires calibration against company data. No audit, penetration test, or certification is reported, and alignment with the information security management standard (International Organization for Standardization, 2022) is structural: the architecture is designed to support compliance with Law Number 27 of 2022 and is not shown to be compliant. The cost conclusions are also calibrated to Indonesian tariffs and labour conditions.

CONCLUSION

This study set out to propose a private cloud design for real estate enterprise data management and to model its cost position with all assumptions explicitly stated. Three contributions answer that aim, each at the level of design and analysis. Architecturally, a workload profile maps real estate application classes to three criticality labels, and propagating the label through storage placement, network tenancy, snapshot cadence, and recovery ordering makes regulatory data classification a property of declared configuration, so divergence between classification and treatment becomes a detectable configuration error. Analytically, simulation of the recovery procedure shows that the thirty-minute target is met with 95 per cent probability only for about 0.9 terabytes of critical data at moderate load and 1.8 terabytes at light load, and in 63 per cent of runs for 1.0 terabyte at moderate load. Economically, a break-even threshold lets an organisation compare the result with its own expenditure; in the reference scenario it is IDR 393 million per year, with a 90 per cent Monte Carlo interval of IDR 272 to 600 million.

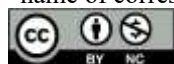
The principal finding for practice is conditional: staffing level and node acquisition cost are the dominant model assumptions, and automation lowers cost only where it produces an actual reduction in staffing expenditure. The principal limitation is that the study reports a proposed design and a model, so neither the recovery estimate nor the security posture has been tested, and the architecture is designed to support compliance rather than shown to be compliant.

Three lines of empirical work follow. The first is a proof-of-concept deployment, on virtualised or small-scale hardware, to measure provisioning latency and sustained reconstruction throughput under concurrent tenant load, replacing the assumed distributions in the recovery simulation. The second is controlled failure injection at instance, single-node, and multi-node scope, with data loss measured against a monotonic write generator and recovery ordering recorded per label. The third is calibration of the cost model and workload profile against procurement quotations, payroll data, and a company's own data inventory. A security and compliance assessment would be required before the architecture could be described as compliant.

ACKNOWLEDGMENT

This research received no external funding and the authors declare no competing interests. The study is analytical and involved no human subjects, no personal data, and no access to any organisation's

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

production systems or financial records. All cost parameters described as indicative, and the workload volumes of Table 2, are the authors' own stated assumptions, not attributed to any organisation. The model and simulation code producing Table 4, the scenario and Monte Carlo results, and Fig. 4 is available from the corresponding author on request.

REFERENCES

- Al-haimi, B., Khalid, H., Zakaria, N. H., & Jasimin, T. H. (2025). Digital transformation in the real estate industry: A systematic literature review of current technologies, benefits, and challenges. *International Journal of Information Management Data Insights*, 5(1), 100340. <https://doi.org/10.1016/j.jjime.2025.100340>
- Banerjee, S., Roy, S., & Khatua, S. (2024). Towards energy and QoS aware dynamic VM consolidation in a multi-resource cloud. *Future Generation Computer Systems*, 157, 376–391. <https://doi.org/10.1016/j.future.2024.03.048>
- Buyya, R., Ilager, S., & Arroba, P. (2024). Energy-efficiency and sustainability in new generation cloud computing: A vision and directions for integrated management of data centre resources and workloads. *Software: Practice and Experience*, 54(1), 24–38. <https://doi.org/10.1002/spe.3248>
- Ciptaningtyas, H. T., Hariadi, R. R., Husni, M., Ghozali, K., Sholikhah, R. W., & Setyadharma, I. M. D. (2023). OpenStack implementation using multinode deployment method for private cloud computing infrastructure. In *Proceedings of the 2023 International Seminar on Intelligent Technology and Its Applications* (pp. 12–17). Institute of Electrical and Electronics Engineers. <https://doi.org/10.1109/ISITIA59021.2023.10221042>
- Goyal, S., & Awasthi, L. K. (2025). Adaptive multi-objective virtual machine consolidation for energy-efficient cloud data centers. *Journal of Grid Computing*, 23, 21. <https://doi.org/10.1007/s10723-025-09808-3>
- Harth, C., Maqbool, R., & Gledson, B. (2025). Augmenting digital transformation toward Real Estate 4.0. *Journal of European Real Estate Research*. <https://doi.org/10.1108/JERER-05-2024-0034>
- Heuchert, S., Rimal, B. P., Reisslein, M., & Wang, Y. (2025). Design of a small-scale and failure-resistant IaaS cloud using OpenStack. *Applied Computing and Informatics*, 21(1/2), 164–183. <https://doi.org/10.1108/ACI-04-2021-0094>
- Hidayat, T., Ramli, K., Harwahu, R., Salman, M., & Gunawan, T. S. (2025). Reinforcement learning-driven hybrid precopy/postcopy VM migration for energy-efficient data centers. *IEEE Access*, 13, 169521–169533. <https://doi.org/10.1109/ACCESS.2025.3612958>
- International Organization for Standardization. (2022). *ISO/IEC 27001:2022 Information security, cybersecurity and privacy protection: Information security management systems, requirements*. ISO.
- Khoda Parast, F., Damghani, S. A., Kelly, B., Wang, Y., & Kent, K. B. (2025). Efficient security interface for high-performance Ceph storage systems. *Future Generation Computer Systems*, 164, 107571. <https://doi.org/10.1016/j.future.2024.107571>
- Lambert, S., Ostapenco, V., Lefevre, L., Caron, E., Orgerie, A.-C., Fichel, B., & Grivel, R. (2026). Revisiting virtual machine consolidation to save resources and energy in heterogeneous production cloud infrastructures. *The International Journal of High Performance Computing Applications*. <https://doi.org/10.1177/10943420251399694>
- Liu, B., Song, J., & Geng, W. (2025). Private cloud deployment model selection for cost efficiency: A business perspective. *Kybernetes*, 54(13), 7357–7379. <https://doi.org/10.1108/K-02-2024-0430>
- Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST Special Publication 800-145). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-145>
- Ministry of Energy and Mineral Resources of the Republic of Indonesia. (2024). *Regulation Number 7 of 2024 on electricity tariffs provided by PT PLN (Persero)*. Ministry of Energy and Mineral Resources.
- Nawrocki, P., & Smendowski, M. (2025). A survey of cloud resource consumption optimization methods. *Journal of Grid Computing*, 23, 5. <https://doi.org/10.1007/s10723-024-09792-0>
- Park, J., Jeong, S., & Yeom, K. (2025). Private cloud bespoke orchestrator: Techniques for constructing and operating bespoke-private cloud virtual machine environments for cloud users. *Journal of Cloud Computing*, 14(1), 34. <https://doi.org/10.1186/s13677-025-00760-x>

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

- Republic of Indonesia. (1997). *Law Number 8 of 1997 concerning Company Documents*. State Gazette of the Republic of Indonesia.
- Republic of Indonesia. (2022). *Law Number 27 of 2022 concerning Personal Data Protection*. State Gazette of the Republic of Indonesia.
- Starr, C. W., Saginor, J., & Worzala, E. (2021). The rise of PropTech: Emerging industrial technologies and their impact on real estate. *Journal of Property Investment & Finance*, 39(2), 157–169. <https://doi.org/10.1108/JPIF-08-2020-0090>
- Suyatno, T., Ferdiana, R., & Widyawan, W. (2025). Disaster recovery strategy: RTO and RPO analysis on on-premises and cloud infrastructure. In *Proceedings of the 2025 International Conference on Advancement in Data Science, E-learning and Information System* (pp. 1–6). Institute of Electrical and Electronics Engineers.
- Swanson, M., Bowen, P., Phillips, A. W., Gallup, D., & Lynes, D. (2010). *Contingency planning guide for federal information systems* (NIST Special Publication 800-34, Rev. 1). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-34r1>
- Syailendra, M. R., & Lie, G. (2024). Personal data protection law in Indonesia. *Indonesian Law Review*, 14(2), 172–186.
- Tagliaro, C., Pome, A. P., Migliore, A., & Danivska, V. (2025). Technology like a fork: How PropTech shapes real estate innovation. *Journal of European Real Estate Research*, 18(1), 4–26. <https://doi.org/10.1108/JERER-05-2024-0035>
- Yao, W., Wang, Z., Hou, Y., Zhu, X., Li, X., & Xia, Y. (2023). An energy-efficient load balance strategy based on virtual machine consolidation in cloud environment. *Future Generation Computer Systems*, 146, 222–233. <https://doi.org/10.1016/j.future.2023.04.014>
- Zhao, L., Hu, G., & Xu, Y. (2024). Educational resource private cloud platform based on OpenStack. *Computers*, 13(9), 241. <https://doi.org/10.3390/computers13090241>
- Zou, J., Wang, K., & Zhang, K. (2024). Perspective of virtual machine consolidation in cloud computing: A systematic survey. *Telecommunication Systems*, 87(2), 257–285. <https://doi.org/10.1007/s11235-024-01184-9>