RACK & POWER



Scaling Power Delivery for AI Infrastructure – Cascade vs Aggregated

Praveen Kumar Velpuri Principal Firmware Engineer, Microsoft

Yang Hsiung Principal Firmware Engineer, Microsoft

What this session addresses

A system-level view of multi-shelf power delivery

WHY NOW

GPU demand is pushing beyond isolated shelf assumptions.

01

THE PROBLEM

Sharing capacity introduces coordination and fault challenges.

02

THE PATTERN

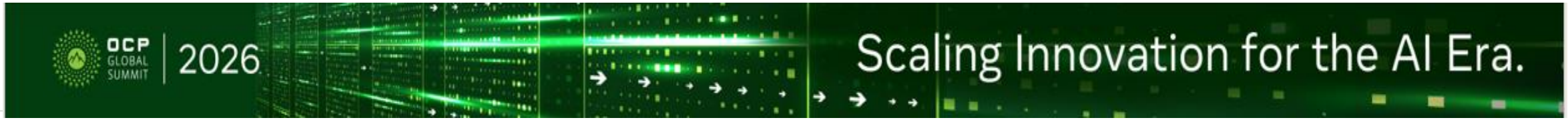
Hierarchical orchestration combines local safety with rack-level policy.

03

THE OUTCOME

Higher usable capacity, lower losses and bounded failures domains.

04



Problem statement

AI racks share capacity, while limits, faults, and transients remain local.

1 USABLE CAPACITY ≠ INSTALLED CAPACITY

Nameplate power is not fully usable.

Redundancy, derating, limits, and reserve define usable capacity.

Model usable capacity

2 SHARED BUS ≠ UNLIMITED SHARING

Aggregation pools power, but shelves and paths retain limits.

A shelf can reach its limit while rack capacity remains available.

Coordinate rack-wide sharing

3 LOCAL FAULTS CAN BECOME RACK EVENTS

Shared paths can expand fault impact.

Isolate the smallest safe domain and preserve healthy capacity.

Contain the fault domain

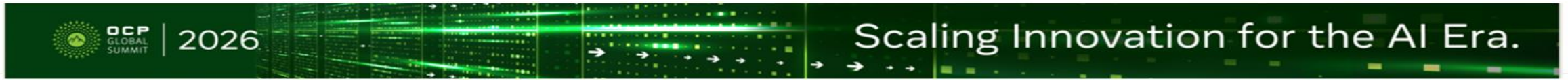
4 GPU TRANSIENTS OUTRUN SOFTWARE

Correlated load steps can consume reserve in milliseconds.

Local protection acts first; rack software re-models, reallocates, and recovers.

Protect fast, recover safely

The challenge: coordinate shared capacity while preserving local protection and predictable recovery.

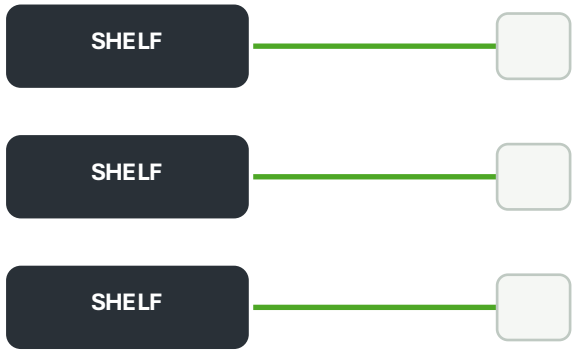


Rack Power Architecture Evolution

Three patterns, three different fault and sharing boundaries

INDEPENDENT

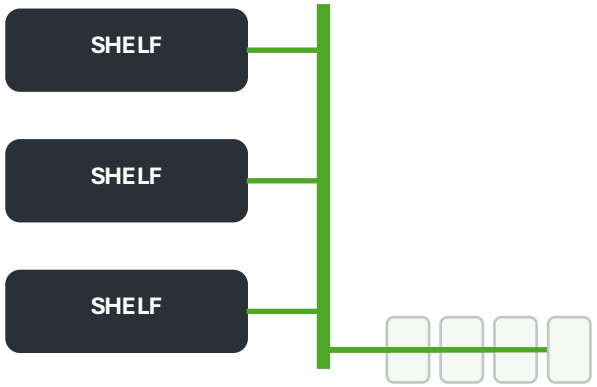
Shelf-local power



Simple isolation
Limited sharing

AGGREGATED

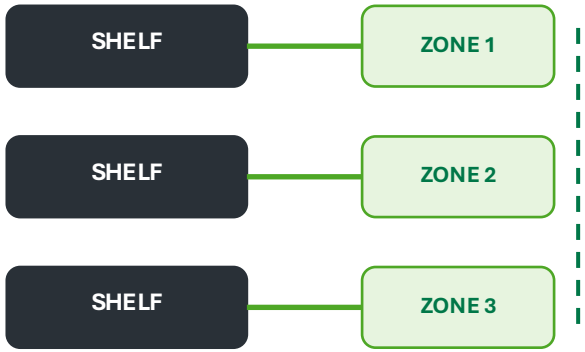
Rack-wide power pool



Maximum pooling
Larger common domain

CASCADED / ZONED

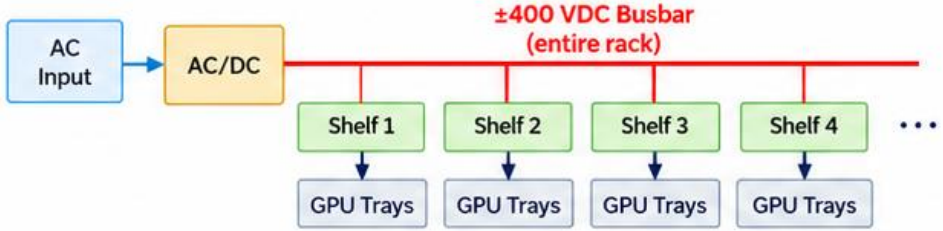
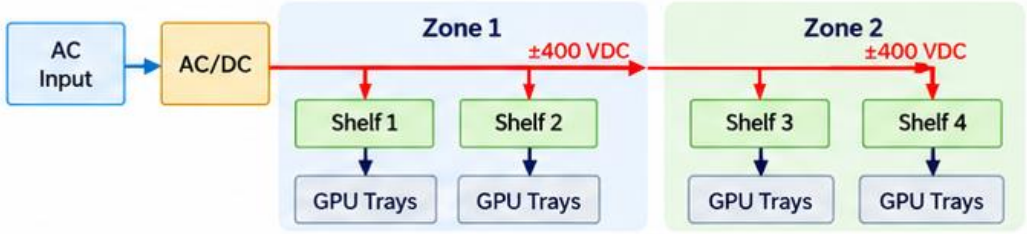
Controlled sharing by domain



Smaller fault domains
More coordination points

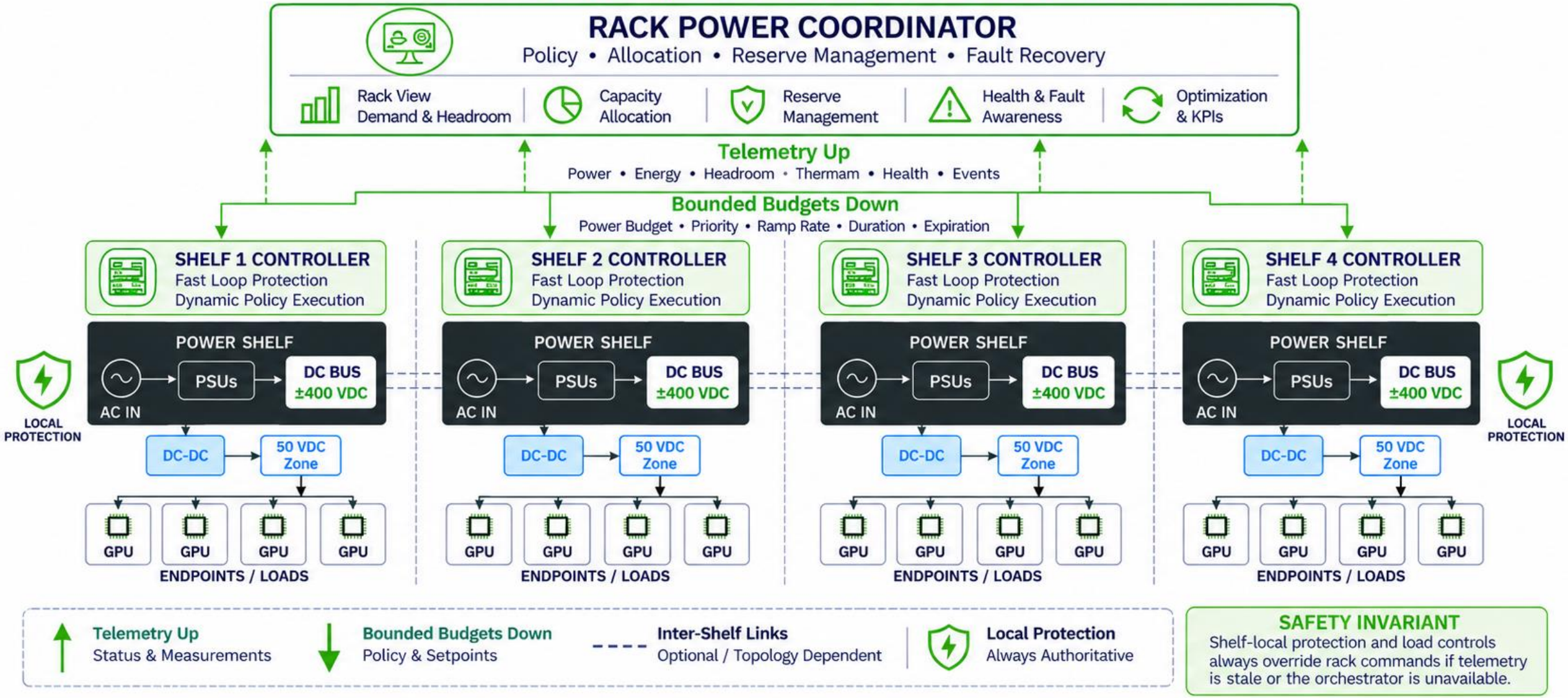
Aggregated vs. Cascaded Power Architecture

Two viable approaches with different trade-offs

Aspect	Aggregated (Single Rack Bus)	Cascaded (Staged Power Zones)
Architecture Overview		
Power Sharing	Whole rack shares one common ±400 VDC busbar. Any shelf can use available rack capacity (within policy).	Each shelf feeds local IT first. Remaining power passes downstream to next shelf via HVDC link.
Capacity Utilization	**High** – minimal stranded capacity, flexible sharing across the rack.	**Good** – local use first, sharing depends on downstream capacity; may have stranded capacity.
Fault Containment	A fault can impact the entire rack busbar (mitigation depends on protection and bus-tie design).	Fault is contained to a shelf/zone; downstream shelves can remain operational.
Isolation / Serviceability	More complex isolation, as shelves share a common bus.	Easier isolation and maintenance at shelf/zone level.
Control Complexity	**Higher** – requires rack-scale coordination and fast protection for the shared bus.	**Lower** – hierarchical control with local shelf autonomy.
Efficiency	Potentially **higher** – better power sharing and load balancing.	**Slightly lower** – possible additional conversion/steering losses and stranded capacity.
Scalability	Simple to scale shelves on the same busbar.	Scales by adding stages/zones; good for very large racks.
Pros and Cons	**Pros** • High rack-wide utilization • Flexible capacity sharing • Simple power distribution (single bus) • Good for balanced loads **Cons** • Larger fault blast radius • More complex protection • Harder isolation and service • Requires rack-scale coordination	**Pros** • Smaller fault domain • Easier isolation and maintenance • Good scalability for large racks • Natural staging and hierarchy • Supports different power zones **Cons** • Potential stranded capacity • Lower overall utilization • More complex power path • Additional conversion/steering loss • Less flexible sharing across zones

Hierarchical Rack Power Coordination

Coordinated Sharing. Local Protection. Higher Utilization.



Manage the Rack Within a Safe Power Envelope

One coordinated loop turns telemetry into a safe power budget; the operating condition determines the response.

1 UNDERSTAND CAPACITY

Observe + Model

Measure power, temperature and health.
Determine current usable capacity and reserve.

2 COORDINATE CAPACITY

Allocate + Verify

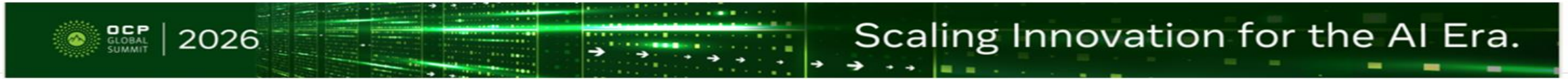
Distribute rack power budgets within the safe envelope.
Verify response and rebalance as conditions change.

3 APPLY THE RIGHT RESPONSE

Condition → Action

Normal: monitor / optimize
Constrained: cap
Fast excursion: local HW throttle
Link degraded: conservative fallback
Faulted: contain

Design principle: rack software coordinates sustained capacity; local hardware protects fast electrical limits.



Contain the Fault, Recover Safely, Prove the Result

Limit fault impact, restore only verified capacity, then measure the outcome.

FAULT → RECOVERY

1 CONTAIN

Detect locally and isolate the smallest safe domain

2 STABILIZE

Confirm voltage, current, and thermal state

3 REALLOCATE

Re-model surviving usable capacity and reset budgets

4 RECOVER

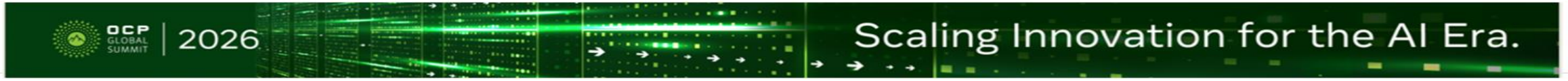
Reconnect gradually only after path and capacity verification

PROVE IT WITH 4 RACK KPIs

USABLE CAPACITY	Safely deliverable power ÷ installed capacity	% / kW
TRANSIENT RESPONSE	Time for protection to limit a fast event	ms
FAULT IMPACT	Rack resources affected by the fault	trays / shelves / zones
RECOVERY	Time to restore stable usable capacity	sec / min

VALIDATE ACROSS **Steady state → GPU load step → Cap change → Link loss / fault → Controlled recovery**

Success = preserve safety, minimize blast radius, retain usable capacity, and return to a stable policy.

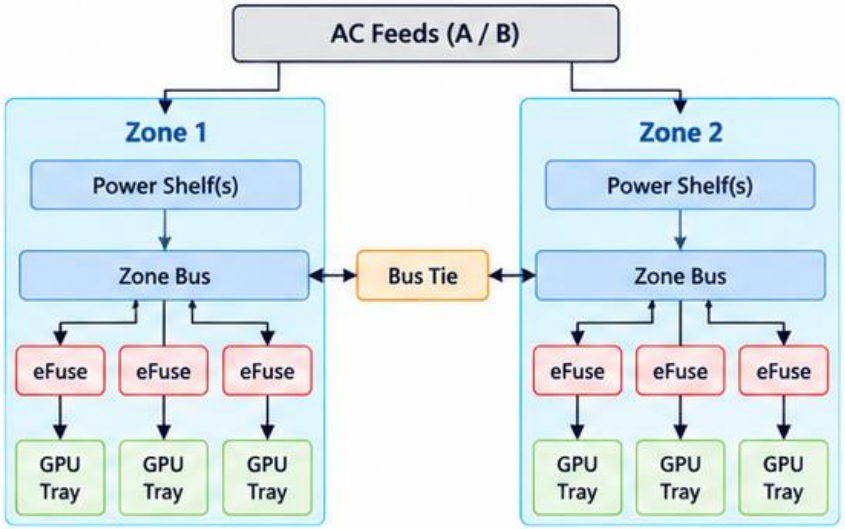


Fault Containment: From Detection to Blast Radius

Protection acts at the smallest available isolation boundary

Fault Scenario	Protection / Response	Result	Blast Radius
1 **1 PSU fails** (within a shelf)	• PSU protection (OCP/OVP/UVP/OTP) • N+2 within shelf • Remaining PSUs carry load	Shelf continues to operate	**0 trays** PSU only
2 **1 shelf fails** (loss of entire shelf)	• Isolate failed shelf • Partner shelf on other feed carries load (N+N at zone)	Zone remains powered	**0 trays** Contained in zone
3 **Full feed lost** (loss of one AC feed)	• Surviving-feed shelf carries zone • Zone-level redundancy (N+N)	Zone continues on redundant feed	**0 trays** Zone-local N+N
4 **Branch fault** (e.g., GFD/OCP/OVP/ UVP/OTP)	• Detect at branch level • Trip affected e-fuse • Healthy branches continue	Other branches unaffected	**1 branch** One GPU tray (or equivalent load)
5 **Bus short in one zone** (zone-bus fault)	• Per-zone isolation • Open bus tie (prevent propagation) • Zone-level protection	Other zones unaffected	**≤ 3 trays** One zone

Power Architecture and Isolation Boundaries



Branch-Level Protection Functions

- **GFD** — Ground fault
- **OCP** — Over current
- **OVP** — Over voltage
- **UVP** — Under voltage
- **OTP** — Over temperature
- **Bus-Tie** — Cross-zone recovery/isolation

Detect locally → Isolate the smallest safe domain → Keep healthy domains operating → Recover only after verification

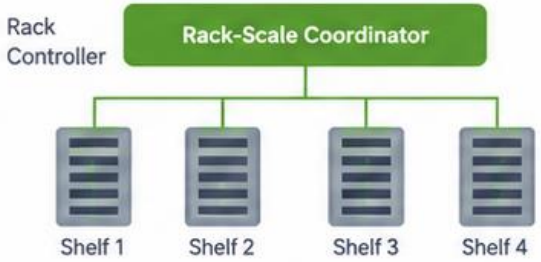
CALL TO ACTION

Turn rack-scale power coordination into a repeatable engineering practice.

1 DESIGN AS A RACK

Treat the rack as one coordinated power system.

Define topology, usable capacity, shared reserve, and fault domains.

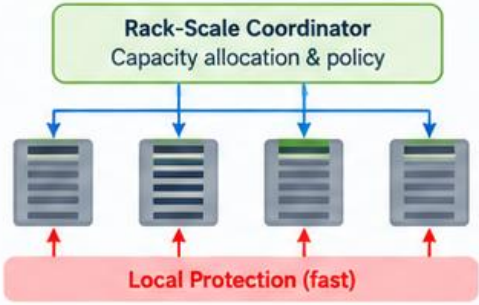


- Model rack-level capacity
- Define reserve and limits
- Identify fault domains

2 COORDINATE & PROTECT

Share capacity at rack scale while keeping fast protection local.

Allocate budgets dynamically; contain faults to the smallest safe domain.



- Dynamic budget allocation
- Shared reserve across shelves/zones
- Fast local protection for transients
- Contain faults to smallest domain

3 MEASURE & PROVE

Validate the architecture under normal, transient, and fault conditions.

Measure usable capacity, stranded capacity, reserve, fault impact, and recovery.

- Usable Capacity & Utilization
- Headroom & Reserve
- Fault Impact & Blast Radius
- Transient Response & Recovery Time

- Define common rack-level KPIs
- Test normal, transient, and fault scenarios
- Verify stability and controlled recovery

4 STANDARDIZE TOGETHER

Create common interfaces, definitions, and metrics through OCP.

Enable interoperable implementations across vendors.

OCP Standards

- Terminology & metrics
- Telemetry & control interfaces
- Operating states & behaviors
- Fault & recovery semantics
- Interoperable management patterns

Optimize at rack scale. | **Protect locally.** | **Measure consistently.** | **Standardize the interfaces.**

Open Discussion

Three Questions for the OCP Community

1 Interfaces

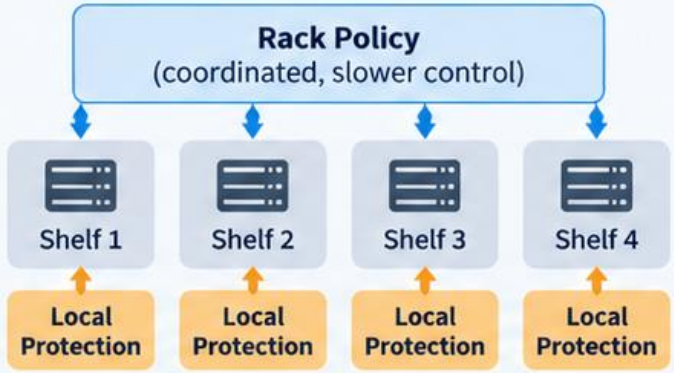
Which minimum **telemetry, capability,** and **control primitives** should be standardized across shelves?



- Common telemetry data model
- Standard control primitives and limits
- Event/fault reporting semantics
- Interoperable management interfaces (e.g., Redfish)

2 Safety boundaries

How should **local protection, rack policy,** and **fallback behavior** be divided and validated?



- Definition of protection boundaries (shelf / zone / rack)
- Required fallback behavior on communication loss
- Validation scenarios for transients and faults
- Criteria for safe recovery and re-enablement

3 Common metrics

Which rack-scale KPIs best demonstrate **utilization, energy savings,** and **resiliency benefits**?



- Common KPI definitions and measurement method
- Before/after comparison across architectures
- Scenarios: normal operation, transient events, faults
- Metrics that enable vendor-to-vendor comparison