
0:00 - Không Phải Lỗi, Đó Là Tính Năng Sinh Tồn
Giải mã giới hạn vật lý của Hệ thống Phân tán qua case-study Shopee lúc nửa đêm.
0:00:00
NotebookLM

Góc Khuất Của Trải Nghiệm “Sập”

Đúng 0h00, bạn gặp phải: Lag, giá nhảy lung tung, giỏ hàng quay đều, báo lỗi “Hệ thống bận”.

Trong mắt người dùng, hệ thống đã “sập”.

Dưới góc nhìn của một Master Engineer, đây không phải là lỗi (bug). Đó là tính năng sinh tồn được thiết kế có chủ đích để cứu hệ thống khỏi cái chết.

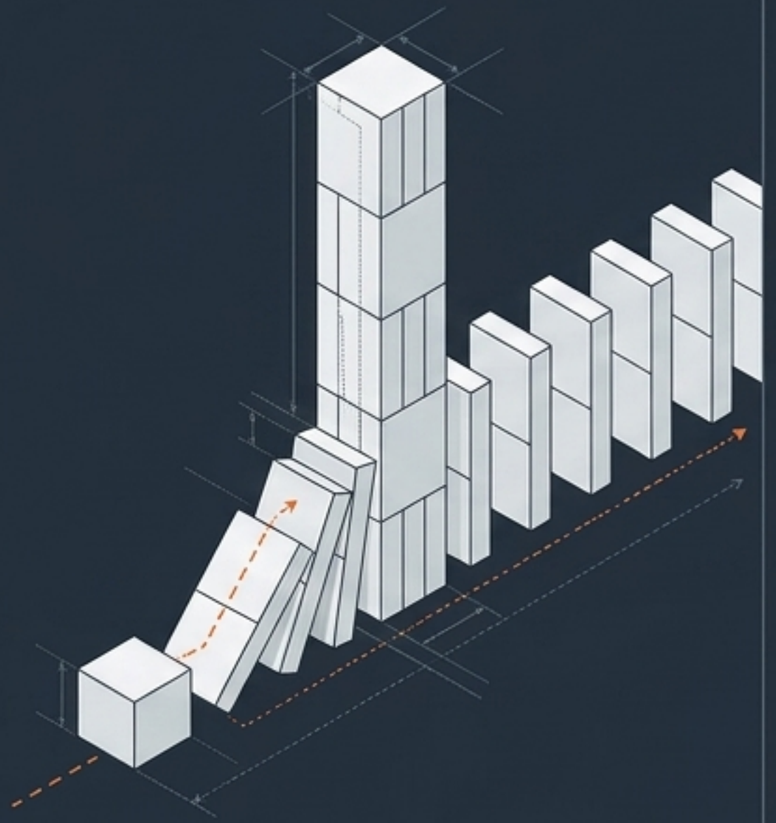


NotebookLM

Giải Phẫu Một Cú “Sập” Khởi Nguyên

Trong hệ thống phân tán, thảm họa hiếm khi do đứt cáp. Nó là hệ quả của các lỗi kiến trúc sau:

Hiệu ứng Domino: Một service nghẽn kéo sập toàn bộ.



Chết ngạt hàng đợi: Request lấp đầy queue, rớt gói tin hàng loạt.



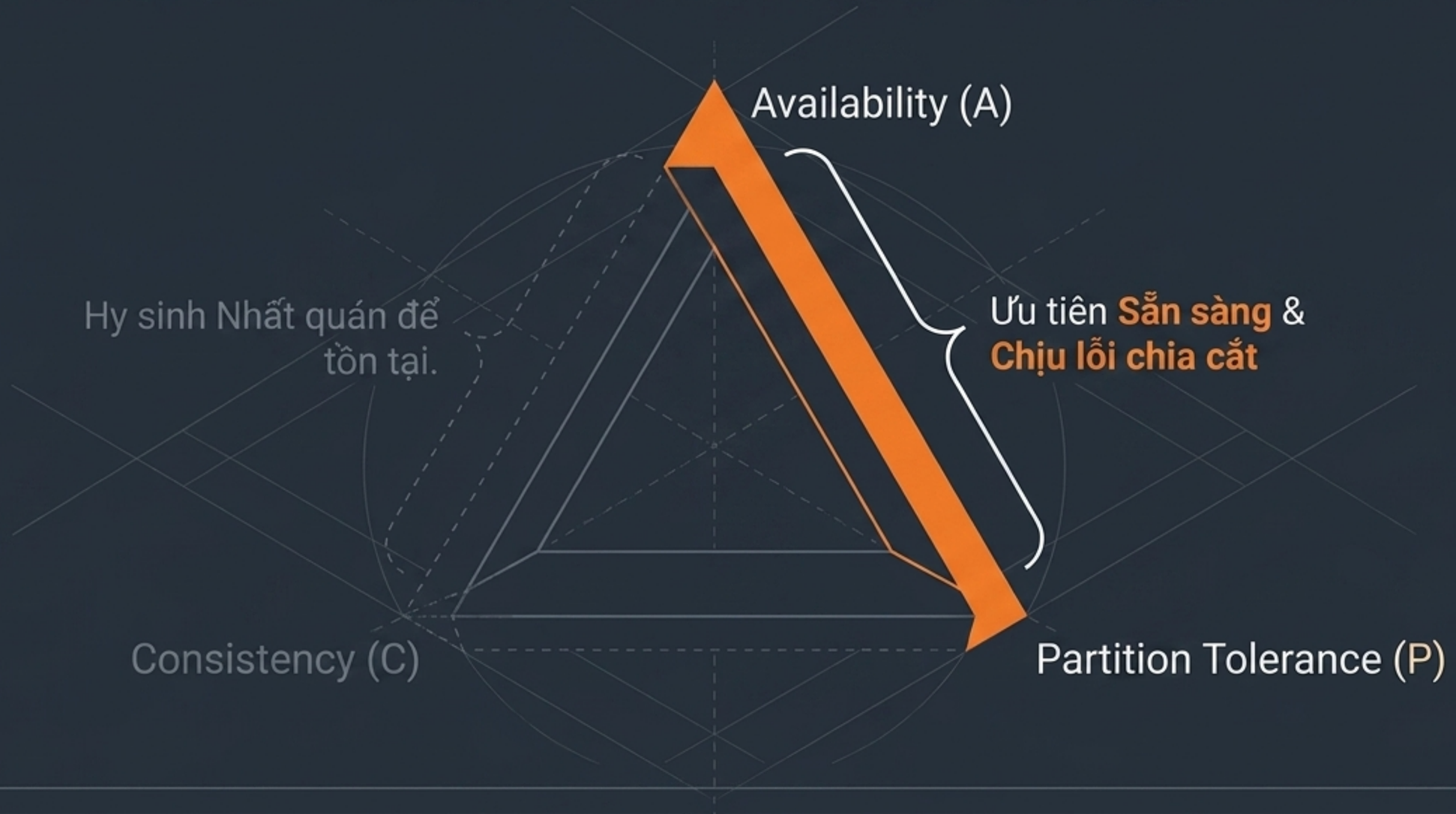
Tắc nghẽn khóa: Dùng Pessimistic Locking ép giao dịch xếp hàng tạo Deadlock.



NotebookLM

Định Lý CAP: Sự Đánh Đổi Tối Thượng

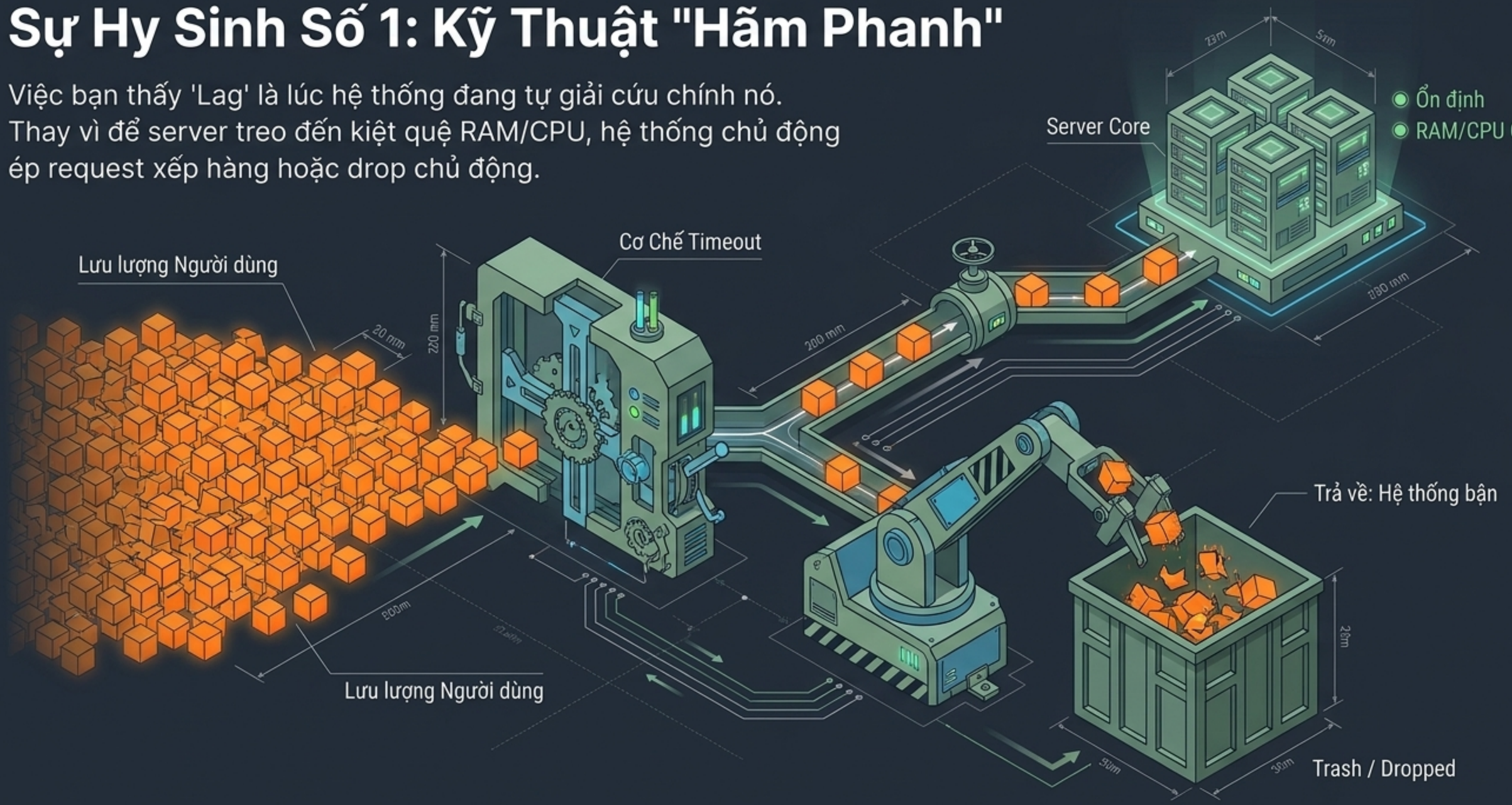
Khi 10 triệu thiết bị cùng gửi request lúc 0h00, hệ thống đối mặt với giới hạn vật lý tuyệt đối. Bạn không thể có tất cả.



NotebookLM

Sự Hy Sinh Số 1: Kỹ Thuật "Hãm Phanh"

Việc bạn thấy 'Lag' là lúc hệ thống đang tự giải cứu chính nó.
Thay vì để server treo đến kiệt quệ RAM/CPU, hệ thống chủ động
ép request xếp hàng hoặc drop chủ động.



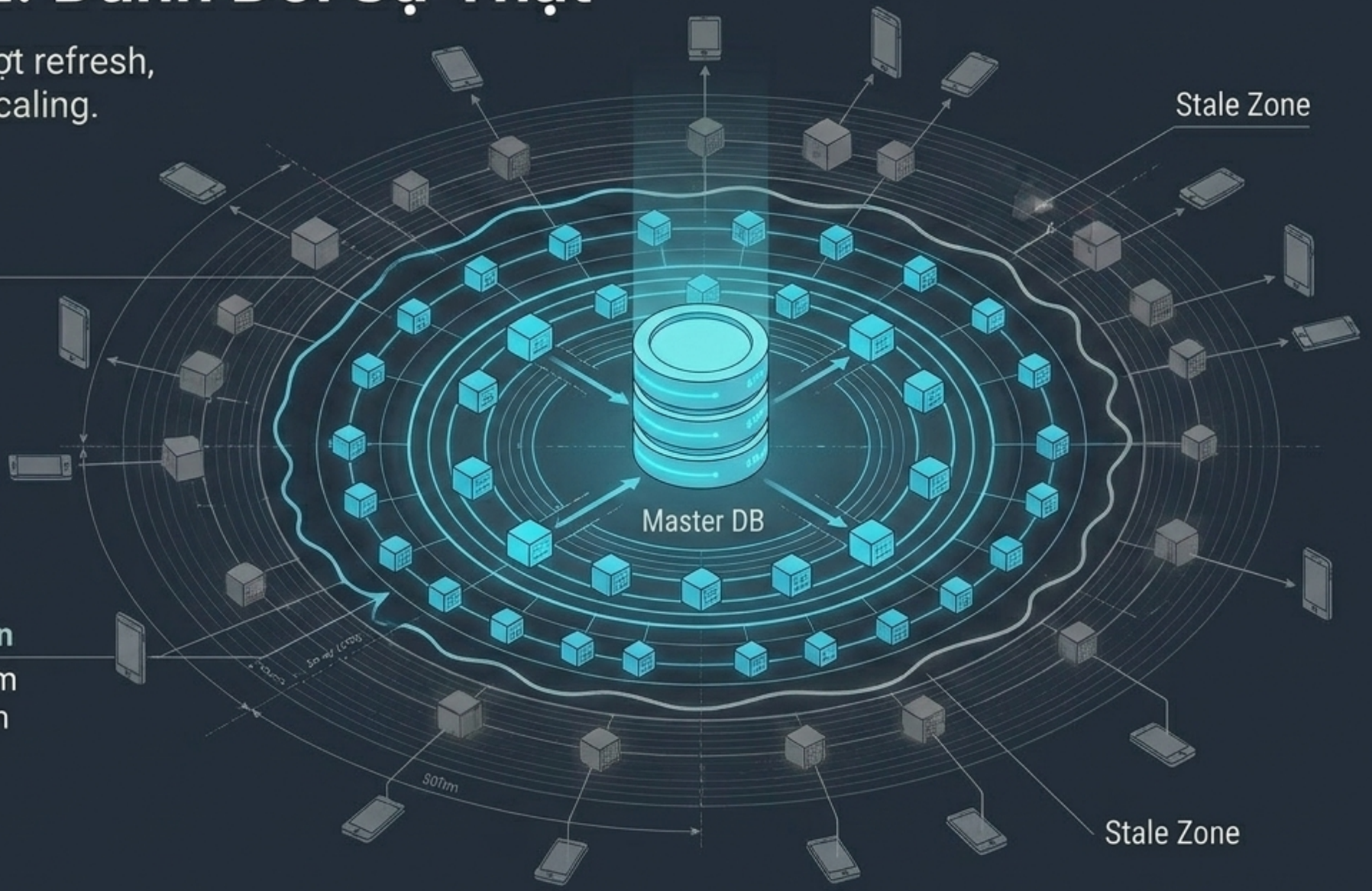
NotebookLM

Sự Hy Sinh Số 2: Đánh Đổi Sự Thật

Để chịu tải hàng chục triệu lượt refresh, hệ thống dùng cơ chế Read Scaling.

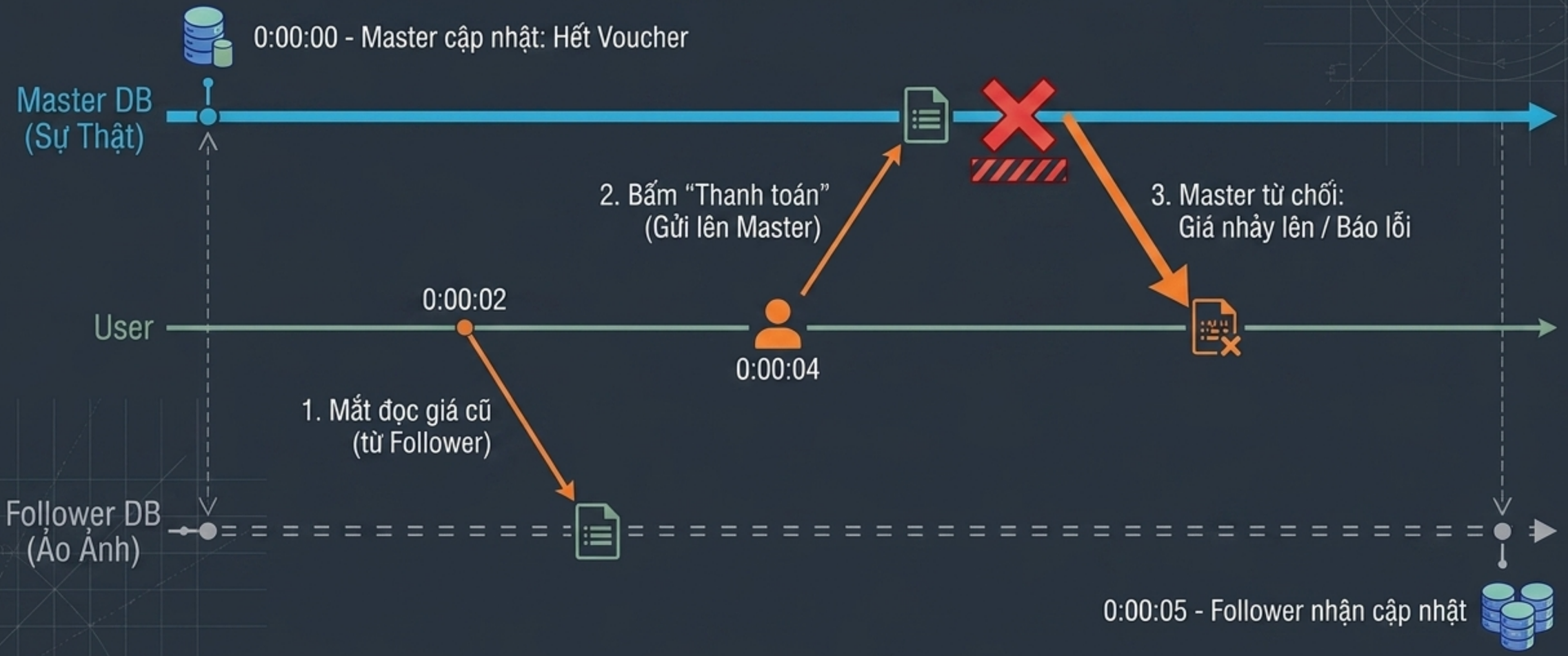
Bản sao
Bản sao (Followers): Dữ liệu được đẩy ra hàng ngàn node phụ.

Độ trễ sao chép (Replication Lag): Lúc tải cao, dữ liệu tạm thời bất đồng bộ, gây ra hiện tượng nhìn thấy giá cũ.



NotebookLM

Giải Mã "Bóng Ma" Dữ Liệu: Giá Nhảy Lung Tung



NotebookLM

Ma Trận Đối Chiếu: Trải Nghiệm vs. Kiến Trúc

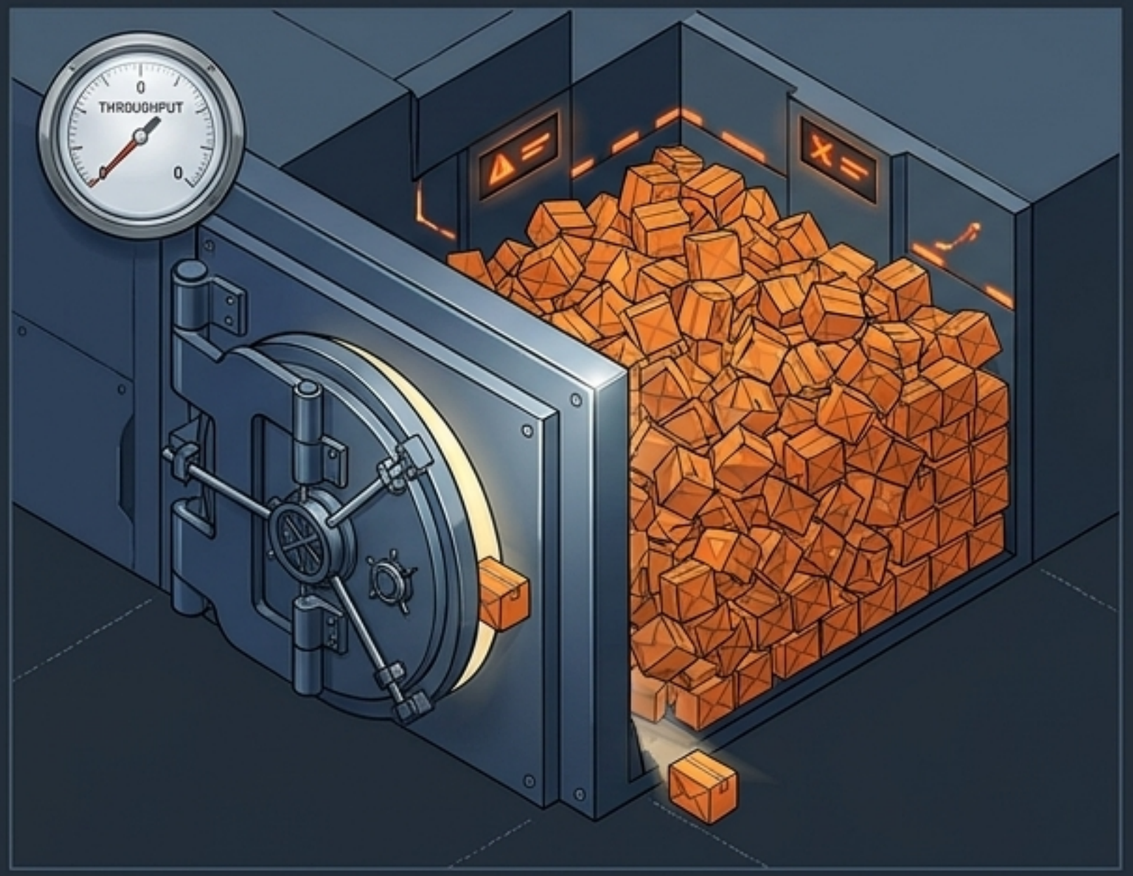
Trải Nghiệm Người Dùng (UX)	Chủ Đích Thiết Kế (Architecture)
App bị treo, màn hình trắng. 	Load shedding & Timeouts (Cứu CPU/RAM). 
Đang thanh toán thì giá tăng vọt.	Eventual Consistency & Master-node validation (Bảo vệ doanh thu).
Tự động hoàn tiền, hủy đơn sau 5 phút.	Compensating Transaction (Tránh Deadlock).

NotebookLM

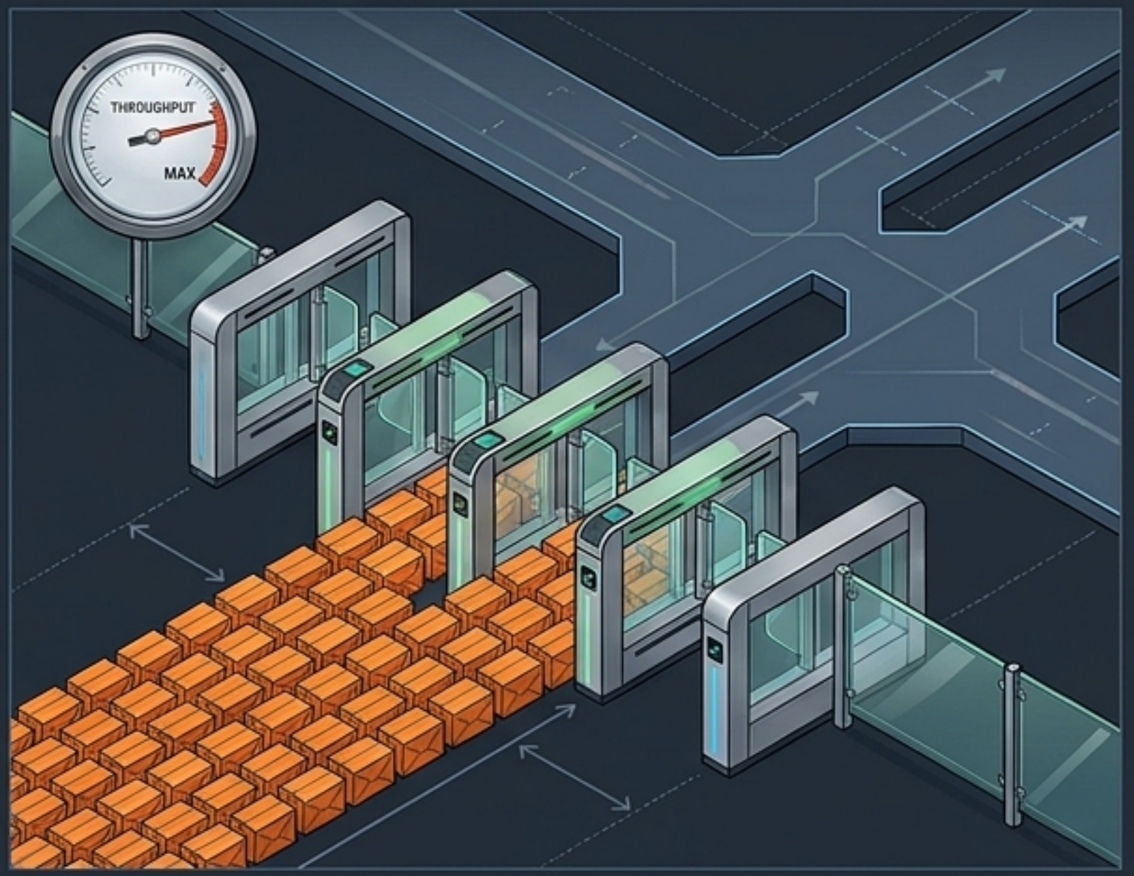
Thế Tiến Thoái Lưỡng Nan Của Khóa

Dùng khóa chéo (Pessimistic Locking) và giao dịch phân tán để ép ACID tuyệt đối sẽ khiến thông lượng (throughput) rơi xuống mức 0 lúc 0h00.

ACID / Pessimistic Locking



BASE / Apology Workflow



NotebookLM

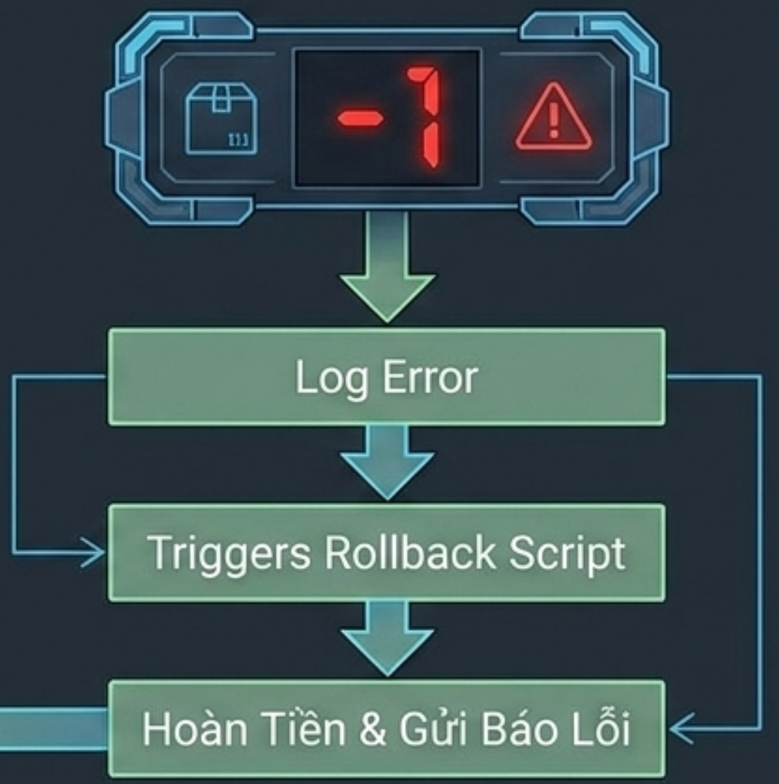
Kỹ Thuật “Quy Trình Xin Lỗi”

Tối đa hóa UX ở mức hiển thị. Ghi nhận giao dịch lỗi ở backend, sau đó xử lý bồi thường.

Ảo ảnh Frontend: Đặt hàng thần tốc



Thực tế Backend: Xử lý bồi thường



NotebookLM

Triết Lý Của Master Engineer

Kỹ thuật hệ thống không phải là theo đuổi sự hoàn hảo, mà là nghệ thuật chọn đúng sự đánh đổi.



Hãm phanh có tính toán để đánh đổi Sự Nhất Quán lấy Thông Lượng và Tính Sẵn Sàng. Hệ thống sống sót để phục vụ 99% khách hàng khác.

NotebookLM