ARCHITECTURE EVOLUTION CASE

From Single-DC Limits to Geographic Active-Active

A real architecture evolution shaped by two constraints: physical capacity had run out, and a single data center had become an unacceptable failure domain.

TRIGGER 01	TRIGGER 02
Beijing IDC capacity was effectively exhausted	**One site still meant one large blast radius**

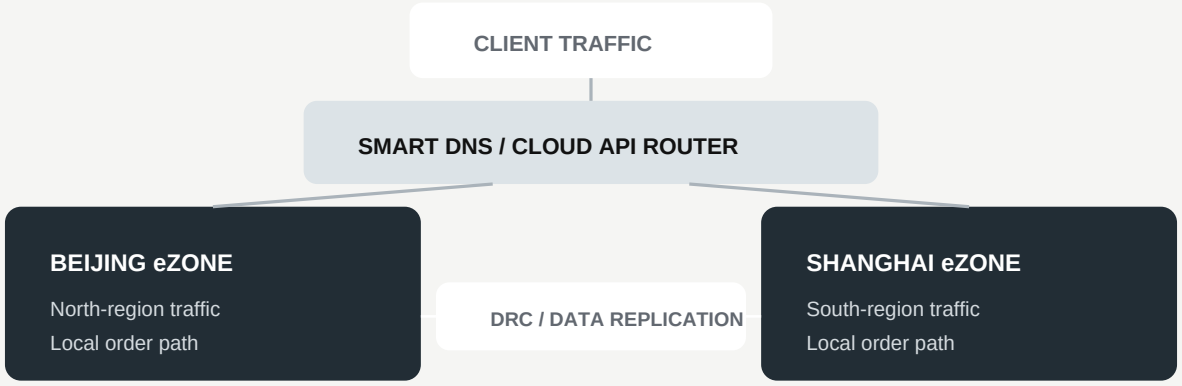
CONTEXT

Ele.me had grown beyond what a single Beijing data center could safely support. Public technical accounts from the period describe the Beijing site as unable to add more physical capacity, while site-level failures remained a material availability risk.

DESIGN PRINCIPLE

The business was naturally regional: a customer, merchant and courier serving one order are usually in the same geography. That made location a useful sharding key and allowed an order journey to remain inside one serving eZone.

SIMPLIFIED ARCHITECTURE - COMPANY-LEVEL CONTEXT



Normal path: route by geography, keep the order journey local, replicate persistent data across sites, and retain the ability to shift traffic when a site fails.

ARCHITECTURE LESSON

Active-active was not "duplicate everything." It required deliberate decisions about locality, failure boundaries, routing, consistency and which dependencies could safely remain global.

MY SCOPE IN THE EVOLUTION

The edge cases were different for mobile and persistent connections

The company-level multi-active architecture was broader than my ownership. My work sat at the mobile access boundary, where stateless HTTP routing and long-lived connections needed different treatment.

MOBILE ACCESS & APM

I participated in the mobile-side design and build-out for the multi-data-center architecture, including access paths, zone-aware behavior and mobile observability/APM.

LONG-LIVED CONNECTIONS

The real-time connection platform was a Layer-4 style problem, not the same Layer-7 HTTP routing problem solved by the cloud API Router. Its failover and routing path had to be designed separately.

WHAT CHANGED AT THE MOBILE / CONNECTION BOUNDARY

DIFFERENT ROUTING PATHS	For normal HTTP traffic, geography-aware routing could happen through the cloud entry/API Router. For the persistent-connection path I worked on, we used intelligent DNS to connect more directly to the serving IDC rather than simply reusing the Layer-7 path.
SERVER-INITIATED PUSH	Push traffic runs in the opposite direction from a client-initiated HTTP request. The system needed a global way to locate which data center currently held a rider or user connection before delivering the message.
LOCAL RUNTIME DEPENDENCIES	The design aimed to keep the operational order path inside one IDC. Cache and messaging were treated as local serving dependencies rather than requiring synchronous cross-site calls.
ZONE PARTICIPATION	I participated in broader design discussions around zone placement and how mobile-facing systems would behave as traffic moved between the Beijing and Shanghai sites.

SCOPE BOUNDARY

I did not own Ele.me's DRC database-replication platform. That layer was built by the data/database architecture team. My responsibility was to make mobile access, real-time connectivity and related systems work correctly within the multi-active architecture.

OPERATIONAL VALIDATION

The design was exercised in real production failovers. Public operations accounts later described multiple whole-site switchovers, including a Shanghai network outage where traffic was moved to Beijing.

PUBLIC CONTEXT REFERENCES

Ele.me multi-active architecture publications and talks: geographic sharding / API Router / eZone design; DRC bidirectional replication; operational multi-active switchovers. Sources include Tencent Cloud Developer Community, Alibaba Cloud Developer Community and published Ele.me operations talks.